L'AKADÉMIE DU DIGITAL · TRAINING PROGRAMME Qualiopi certification being prepared

Développement Web Full-Stack

FOR-DEV-M2-A · Development, DevOps & Data

Concevoir, développer et déployer une application web complète, du front-end au back-end

DURATION
42 heures — 6 jours

PRICE
2?940 € HT

AUDIENCE
Schools & training centres

ACCESS
Access within 2 working days after validation of needs and prerequisites.

Tarif établi sur la base de 42h à 70 € HT/h. Parcours dédié aux écoles et centres de formation, établi sur devis ; tarif indicatif et négociable selon le volume et le contexte.

Le module alterne séquences théoriques et ateliers pratiques tout au long de chaque journée. Le panorama théorique compare les principales stacks (front : React, Vue.js, Angular, Svelte ; back : Node.js/Express, PHP/Laravel, PHP/Symfony) ; les ateliers suivent un fil rouge React + Node.js/Express, avec quelques exemples comparatifs Vue.js. La stack reste adaptable au besoin.

Learning objectives

- Maîtriser les fondamentaux du front-end : HTML, CSS, JavaScript et un framework moderne (React)
- Situer et comparer les principaux frameworks front (React, Vue.js, Angular, Svelte) et back (Node.js/Express, PHP/Laravel, PHP/Symfony)
- Comprendre l'architecture client-serveur et le rôle des API
- Développer un back-end et une API REST (Node.js/Express)
- Concevoir et interroger une base de données (relationnelle et/ou NoSQL)
- Connecter front et back en une application full-stack fonctionnelle, puis la déployer et la sécuriser

Target audience

- Étudiants et apprenants des écoles et centres de formation partenaires
- Développeurs juniors ou en reconversion souhaitant maîtriser la chaîne full-stack
- Profils ayant les bases de la programmation (module Fondamentaux conseillé)

Prerequisites

Bases de la programmation requises (logique, variables, fonctions, structures de données) — le module « Programmation — Fondamentaux multi-langages » est fortement recommandé. Une aisance avec un éditeur de code et Git est attendue. Ces éléments sont validés par un formulaire en ligne et / ou un entretien préalable.

- Maîtriser les bases de la programmation (variables, conditions, boucles, fonctions, structures de données).

- Avoir des notions de JavaScript (module Fondamentaux conseillé).
- Savoir utiliser un éditeur de code et les bases de Git.
- Disposer d'une logique de résolution de problèmes et d'une autonomie de travail.

Programme

Jour 1 — Le web & le front-end (HTML/CSS)

Fonctionnement du web	1h00	Client/serveur, HTTP, navigateur, structure d'un projet web. Exposé interactif, schémas, exemples de requêtes.	Questions orales de compréhension.
Page HTML sémantique	1h30	Intégrer une page HTML structurée et sémantique. TP guidé, gabarit de départ, accompagnement.	Page HTML validée.
CSS & mise en page moderne	1h00	Flexbox, grid, responsive design. Exposé, démonstration, exemples responsive.	QCM sur le CSS moderne.
Styliser & rendre responsive	2h30	Styliser et adapter une interface à toutes tailles d'écran. TP, design fourni, tests multi-écrans.	Interface responsive validée.

Jour 2 — JavaScript côté client & interactivité

JavaScript & DOM	1h15	Manipulation du DOM, événements, dynamisme. Exposé, démonstration en direct, exemples.	Questions orales ; QCM.
Page interactive	2h15	Rendre une page interactive (événements, mises à jour). TP guidé, exercices progressifs.	Interactions validées.
Requêtes asynchrones & API	1h00	fetch, promesses, consommation d'une API. Exposé, schéma requête/réponse, exemples.	QCM sur l'asynchrone.
Consommer une API	2h30	Afficher des données depuis une API publique. TP, API publique, gestion de l'affichage.	Affichage des données validé.

Jour 3 — Front-end moderne (React, comparatif Vue.js)

Panorama des frameworks front	1h15	React, Vue.js, Angular, Svelte : philosophies, courbe d'apprentissage, écosystème, cas d'usage. Exposé comparatif, tableau de comparaison, exemples React vs Vue.js.	Questions orales ; QCM sur le choix de framework.
Composants React	2h30	Construire des composants (props, state) ; exemples comparatifs Vue.js. TP guidé React, extraits comparatifs Vue.js, accompagnement.	Composants fonctionnels validés.
État & routes	1h00	Gestion de l'état et navigation entre vues. Exposé, démonstration, schéma de l'état.	QCM sur l'état et le routage.
Mini-application front	2h15	Assembler une mini-application React. TP, projet front guidé, revue collective.	Mini-app front validée.

Jour 4 — Back-end & API REST (Node.js/Express)

Panorama back & architecture serveur	1h15	Node.js/Express, PHP/Laravel, PHP/Symfony ; architecture, API REST. Exposé comparatif, tableau des stacks back, schéma REST.	Questions orales ; QCM sur les stacks back.
Première API	2h15	Créer des routes et une première API (Node.js/Express). TP guidé, environnement préparé, accompagnement.	Routes API fonctionnelles.
CRUD, middlewares & erreurs	1h00	Opérations CRUD, middlewares, gestion des erreurs. Exposé, exemples, bonnes pratiques.	QCM sur le CRUD.
API REST complète	2h30	Construire une API REST CRUD complète. TP, tests des endpoints (client API), correction collective.	API CRUD validée.

Jour 5 — Base de données & intégration full-stack

Bases de données SQL & NoSQL	1h15	Relationnel (SQL) et NoSQL ; modélisation des données. Exposé, schémas de modèles, comparaison SQL/NoSQL.	Questions orales ; QCM.
Créer & interroger une base	2h15	Créer une base et écrire des requêtes. TP guidé, jeu de données, requêtes progressives.	Requêtes validées.
Intégration des couches	1h00	Connecter back ? base et front ? API. Exposé, schéma d'architecture full-stack.	QCM sur l'intégration.
Relier front, API et base	2h30	Faire communiquer les trois couches. TP d'intégration, débogage, revue collective.	Chaîne full-stack fonctionnelle.

Jour 6 — Authentification, déploiement & projet

Authentification & sécurité	1h00	Authentification, variables d'environnement, validation, bonnes pratiques. Exposé, exemples, rappels de sécurité.	QCM sur la sécurité de base.
Ajouter une authentification	2h00	Mettre en place une authentification simple. TP guidé, scénario d'authentification.	Authentification fonctionnelle.
Déploiement & CI/CD léger	1h00	Hébergement, mise en ligne, intégration continue légère, Git. Exposé, démonstration de déploiement.	QCM sur le déploiement.
Déploiement du projet & restitution	2h30	Déployer l'application full-stack et la présenter. TP de déploiement, présentations, revue collective, QCM final, enquête de satisfaction.	Évaluation sommative : application déployée + QCM final.

Teaching methods and resources

In-person training at our premises or yours, in small groups. A mix of theory, hands-on workshops and practical scenarios.

Required equipment :

16 Go de Ram recommandés

- Les stagiaires doivent disposer d'un PC avec le Wifi et 16 Go de RAM recommandés.
- Effectif adapté au format école / centre de formation.
- Parcours dédié aux écoles et centres de formation, établi sur devis.

Assessment methods

Learning outcomes are assessed during and at the end of the training; an attendance certificate is given to every learner.

Accessibility

Training accessible to people with disabilities. Contact us before registering so we can study the adjustments needed.

Funding

Quote or training agreement on request. Qualiopi certification is being prepared: eligibility for OPCO funding will open once it is obtained.

Certification audit planned for T1 2027. No OPCO eligibility before it is obtained.